UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/847,813	05/01/2001	Curt Wohlgemuth	OMNI0008	6351

7590 03/16/2007

PERKINS COIE LLP
ATTN: Mr. Brian R. Coleman
101 Jefferson Drive
Menlo Park, CA 94025

EXAMINER	
LANIER, BENJAMIN E	
ART UNIT	**PAPER NUMBER**
2132	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	03/16/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	**Application No.** 09/847,813	**Applicant(s)** WOHLGEMUTH ET AL.	
	Examiner Benjamin E Lanier	**Art Unit** 2132	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *05 March 2007*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-3,10-12,19,25,35-37 and 40-48* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-3,10-12,19,25,35-37 and 40-48* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

Response to Amendment

1. Applicant's amendment filed 05 March 2007 amends claim 1. Claims 31-34, 38, and 39 have been cancelled. Claims 45-58 have been added. Applicant's amendment has been fully considered and entered.

Response to Arguments

2. Applicant's arguments filed 05 March 2007 have been fully considered but they are not persuasive. Applicant argues that Vinson et al. is not 102(e) prior art because Vinson et al. "was not granted a patent prior to the invention by the applicant." This argument is not persuasive because 102(e) requires that "the invention was described in … (2) a patent grated on an application for patent by another **FILED** in the United States before the invention by the applicant for patent." Vinson et al. is clearly 102(e) prior art since the granted patent (US 6,453,334) was filed for on 16 June 1998, which is before applicant's filing date of 01 May 2001.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 1-3, 10-12, 19, 25, 35-37, 40-48 are rejected under 35 U.S.C. 102(e) as being anticipated by Vinson, U.S. Patent No. 6,453,334. Referring to claims 1, 10, Vinson discloses a

method and apparatus to allow remotely located computer programs to be accessed on a local computer using a network file system that simulates a local drive on a client computer (Col. 1, lines 13-24 & Col. 2, lines 37-43), which meets the limitation of providing a network file system on a client. The user uses their web browser to navigate a web site, and clicks on link indicating a target program listed on a web page (Col. 5, lines 40-42). The link points to the index file for that target program. (Col. 5, lines 42-43). The web browser initiates retrieval of the index file, and based on the MIME type for the index file, knows that the index file should be downloaded to the client machine and the client agent started with the location of the index file given as an argument to the client agent (Col. 5, lines 43-52). When authenticated the FSD will add a newly created process to its list of processes that can access the files referenced by the index file (Col. 7, lines 28-37). All file operations are handled by the FSD, which downloads, caches, decompresses, and decrypts the pieces of the program as needed (Col. 7, lines 47-50), which meets the limitation of wherein said network file system handles and forwards requests from streaming enabled local processes on said client that are directed at streaming software files located on said server. A deathwatch thread waits for a timeout when the time allowed for the process to access the program expires (Col. 8, lines 22-25), which meets the limitation of wherein said network file system examines said requests, and either grants or denies each of said requests depending on whether the request is justifiable from a security perspective by using information such as the history of previous access by the streaming enabled process. Furthermore, requests for access to the program are examined to see if the current process ID associated with the request is not in the process access list in the specified program descriptor block, and access is denied (Col. 14, lines 33-37), which meets the limitation of the nature of the originating streaming enabled process. For requests that contain a path, access will be granted to each top level directory in which the corresponding program descriptor block contains the current process ID in it's process access list (Col. 14, lines 42-46), which meets the limitation of providing a network redirector component of said network file system. Requests that do not contain a path are handled on a case-by-case manner (Col. 14, lines 52-53), which meets the limitation of wherein said network redirector component makes visible to said network file system, a path that represents the server where said streaming software files are stored.

Referring to claims 2, 11, Vinson discloses that the FSD is called via a dispatch routine, which indicates that the newly created process is to be given access to the target program specified by the index file (Col. 7, lines 28-31), which meets the limitation of said network file system registers dispatch routines with the client operating system that handle zero or more common file operations selected from the group consisting of open, read, write, and close; wherein a dispatch routine examines a file request and decides whether to grant or deny said file request. All operations are handled by the FSD, which downloads, caches, decompresses, and decrypts the pieces of the program as needed (Col. 7, lines 47-49 & Col. 13, lines 41-59), which meets the limitation of if said file request is granted then said dispatch routine forwards said file request to said server and sends back said server's response to said client operating system.

Referring to claims 3, 12, Vinson discloses that the user uses their web browser to navigate a web site, and clicks on link indicating a target program listed on a web page (Col. 5, lines 40-42). The link points to the index file for that target program. (Col. 5, lines 42-43). The web browser initiates retrieval of the index file, and based on the MIME type for the index file, knows that the index file should be downloaded to the client machine and the client agent started with the location of the index file given as an argument to the client agent (Col. 5, lines 43-52), which meets the limitation of when a local streaming enabled process on said client makes a file request for a streaming software file on said server. Dispatch routines are used by the FSD to receive information about the requested target file (Col. 6, lines 3-58), which meets the limitation of said client operating system calls a dispatch routine with said file request.

Referring to claims 19, 25, Vinson discloses a method and apparatus to allow remotely located computer programs to be accessed on a local computer using a network file system that simulates a local drive on a client computer (Col. 1, lines 13-24 & Col. 2, lines 37-43), which meets the limitation of providing a network file system on a client. The user uses their web browser to navigate a web site, and clicks on link indicating a target program listed on a web page (Col. 5, lines 40-42). The link points to the index file for that target program. (Col. 5, lines 42-43). The web browser initiates retrieval of the index file, and based on the MIME type for the index file, knows that the index file should be downloaded to the client machine and the client agent started with the location of the index file given as an argument to the client agent (Col. 5, lines 43-52). When authenticated the FSD will add a newly created process to its list of processes that can access the files referenced by the index file (Col. 7, lines 28-37). All file operations are handled by the FSD, which downloads, caches, decompresses, and decrypts the pieces of the program as needed (Col. 7, lines 47-50), which meets the limitation of wherein said network file system handles and forwards requests from streaming enabled local processes on said client that are directed at streaming software files located on said server. A deathwatch thread waits for a timeout when the time allowed for the process to access the program expires (Col. 8, lines 22-25), which meets the limitation of wherein said network file system examines said requests, and either grants or denies each of said requests depending on whether the request is justifiable from a security perspective by using information such as the history of previous access by the streaming enabled process. Furthermore, requests for access to the program are examined to see if the current process ID associated with the request is not in the process access list in the specified program descriptor block, and access is denied (Col. 14, lines 33-37), which meets the limitation of the nature of the originating streaming enabled process. Vinson discloses that the FSD is called via a dispatch routine, which indicates that the newly created process is to be given access to the target program specified by the index file (Col. 7, lines 28-31), which meets the limitation of said network file system registers dispatch routines with the client operating system that handle zero or more common file operations selected from the group consisting of open, read, write, and close; wherein a dispatch routine examines a file request and decides whether to grant or deny said file request. All operations are handled by the FSD, which downloads, caches, decompresses, and decrypts the pieces of the program as needed (Col. 7, lines 47-49 & Col. 13, lines 41-59), which meets the limitation of a dispatch routine examines a file request and decides whether to grant or deny said file request, and wherein if said file request is granted, then said dispatch routine allows the requested operation to proceed.

Referring to claim 40, Vinson discloses that all file operations are handled by the FSD, which downloads, caches, decompresses, and decrypts the pieces of the program as needed (Col. 7, lines 47-50), which meets the limitation of wherein said network file system handles and forwards requests from streaming enabled local processes on said client that are directed at streaming software files located on said server. A deathwatch thread waits for a timeout when the time allowed for the process to access the program expires (Col. 8, lines 22-25). Furthermore, requests for access to the program are examined to see if the current process ID associated with the request is not in the process access list in the specified program descriptor block, and access is denied (Col. 14, lines 33-37). For requests that contain a path, access will be granted to each top level directory in which the corresponding program descriptor block contains the current process ID in it's process access list (Col. 14, lines 42-46), which meets the limitation of providing information relating to one or more remote locations where streaming software files are stored, determining whether an originating process that is making said requests for access is a trusted process, whether a history of previous requests for access made by said originating process exhibits a pre-determined pattern of piracy, and whether a section of said streaming software files that is being requested is a critical section that requires protection from piracy.

Referring to claims 35-37, Vinson discloses a method and apparatus to allow remotely located computer programs to be accessed on a local computer using a network file system that simulates a local drive on a client computer (Col. 1, lines 13-24 & Col. 2, lines 37-43). The user uses their web browser to navigate a web site, and clicks on link indicating a target program listed on a web page (Col. 5, lines 40-42). The link points to the index file for that target program. (Col. 5, lines 42-43). The web browser initiates retrieval of the index file, and based on the MIME type for the index file, knows that the index file should be downloaded to the client machine and the client agent started with the location of the index file given as an argument to the client agent (Col. 5, lines 43-52). When authenticated the FSD will add a newly created process to its list of processes that can access the files referenced by the index file (Col. 7, lines 28-37). All file operations are handled by the FSD, which downloads, caches, decompresses, and decrypts the pieces of the program as needed (Col. 7, lines 47-50), which meets the limitation of a processing device for processing a request for access to streaming software files stored on at least one server system that is remote from said processing device. A deathwatch thread waits for a timeout when the time allowed for the process to access the program expires (Col. 8, lines 22-25), which meets the limitation of wherein said processing device comprises a component that determines whether to grant requests for access to said streaming software files based on whether an originating process that is making said requests for access is a trusted process, whether a history of previous requests for access made by said originating process exhibits a pre-determined pattern of piracy, and whether a section of said streaming software files that is being requested is a critical section that requires protection from piracy. Furthermore, requests for access to the program are examined to see if the current process ID associated with the request is not in the process access list in the specified program descriptor block, and access is denied (Col. 14, lines 33-37). For requests that contain a path, access will be granted to each top level directory in which the corresponding program descriptor block contains the current process ID in it's process access list (Col. 14, lines 42-46). Requests that do not contain a path are handled on a case-by-case manner (Col. 14, lines 52-53), which meets the limitation of a redirector component that is associated with said processing device for informing said processing device of one or more locations in which said streaming software files are stored.

Referring to claim 41, Vinson discloses that all file operations are handled by the FSD, which downloads, caches, decompresses, and decrypts the pieces of the program as needed (Col. 7, lines 47-50), which meets the limitation of wherein said network file system handles and forwards requests from streaming enabled local processes on said client that are directed at streaming software files located on said server. A deathwatch thread waits for a timeout when the time allowed for the process to access the program expires (Col. 8, lines 22-25). Furthermore, requests for access to the program are examined to see if the current process ID associated with the request is not in the process access list in the specified program descriptor block, and access is denied (Col. 14, lines 33-37). For requests that contain a path, access will be granted to each top level directory in which the corresponding program descriptor block contains the current process ID in it's process access list (Col. 14, lines 42-46). Vinson discloses that the FSD is called via a dispatch routine, which indicates that the newly created process is to be given access to the target program specified by the index file (Col. 7, lines 28-31), which meets the limitation of means for examining a request for access to said streaming software files, and means for determining whether said requests can be granted based on whether an originating process that is making said requests for access is a trusted process, that a history of previous requests for access made by said originating process lacks a pre-determined pattern of piracy or that a section of said streaming software files that is being requested is a non-critical section, a means for forwarding said request to a corresponding remote server that is responsible for serving said streaming software files.

Referring to claim 42-44, Vinson discloses that all file operations are handled by the FSD, which downloads, caches, decompresses, and decrypts the pieces of the program as needed (Col. 7, lines 47-50). A deathwatch thread waits for a timeout when the time allowed for the process to access the program expires (Col. 8, lines 22-25). Furthermore, requests for access to the program are examined to see if the current process ID associated with the request is not in the process access list in the specified program descriptor block, and access is denied (Col. 14, lines 33-37). For requests that contain a path, access will be granted to each top level directory in which the corresponding program descriptor block contains the current process ID in it's process access list (Col. 14, lines 42-46), which meets the limitation of providing information relating to one or more remote locations where streaming software files are stored, receiving a request from a computer process for access to said streaming software files, determining if a trusted process/history of previous requests for access made by said computer process lacks a pre-determined pattern of piracy/critical section. All operations are handled by the FSD, which downloads, caches, decompresses, and decrypts the pieces of the program as needed (Col. 7, lines 47-49 & Col. 13, lines 41-59), which meets the limitation of if trusted process/history of previous requests of said computer process lacks a pre-determined pattern of piracy/critical section, then forwarding said request to a corresponding remote server that is responsible for serving said streaming software files.

Referring to claim 45, Vinson discloses that requests for access to the program are examined to see if the current process ID associated with the request is not in the process access list in the specified program descriptor block, an access is denied (Col. 14, lines 33-37), which meets the limitation of using a filtering mechanism that is associated with the client for filtering requests for access to the streaming software files.

Referring to claim 46, Vinson discloses that requests that contain a path, access will be granted to each top level directory in which the corresponding program descriptor block contains the current process ID in it's process access list (Col. 14, lines 42-46), which meets the limitation of providing information relating to one or more remote locations where streaming software files are stored.

Referring to claim 47, Vinson discloses that requests that contain a path, access will be granted to each top level directory in which the corresponding program descriptor block contains the current process ID in it's process access list (Col. 14, lines 42-46), which meets the limitation of providing information relating to one or more remote locations, including the server, where streaming software files are stored. Vinson discloses that the FSD is called via a dispatch routine, which indicates that the newly created process is to be given access to the target program specified by the index file (Col. 7, lines 28-31), which meets the limitation of using dispatch routines for examining a request for access to said streaming software files, after examining said request and if it is determined that a history of previous requests for access made by said originating process lacks a pre-determined pattern of piracy or that a section of said streaming software files that is being requested is a non-critical section, then forwarding said request to the server.

Referring to claim 48, Vinson discloses that all file operations are handled by a client side FSD (Col. 3, lines 25-26), which meets the limitation of using a filtering mechanism on the client for filtering requests for access to streaming software files. Remotely located computer programs can be accessed on a local computer using a network file system that simulates a local drive on a client computer (Col. 1, lines 13-24 & Col. 2, lines 37-43), which meets the limitation of using a revealing mechanism to reveal to said client one or more remote locations, including the server, on which said requested streaming software files are stored.

Conclusion

5. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Benjamin E. Lanier whose telephone number is 571-272-3805. The examiner can normally be reached on M-Th 7:30am-5:00pm, F 7:30am-4pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Gilberto Barron can be reached on 571-272-3799. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Benjamin E. Lanier

GILBERTO BARRON JR
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100